

# **Отчёт по лабораторной работе 13**

## **Настройка NFS**

Авдадаев Джамал Геланиевич

# Содержание

<b>1 Введение</b>	<b>5</b>
1.1 Цель работы . . . . .	5
<b>2 Процесс работы</b>	<b>6</b>
2.1 Настройка сервера NFSv4 . . . . .	6
2.2 Монтирование NFS на клиенте . . . . .	9
2.3 Подключение каталогов к дереву NFS . . . . .	11
2.4 Подключение каталогов для работы пользователей . . . . .	13
2.5 Внесение изменений в настройки внутреннего окружения виртуальных машин . . . . .	17
<b>3 Итоги</b>	<b>19</b>
3.1 Вывод . . . . .	19
3.2 Контрольные вопросы . . . . .	19

# Список иллюстраций

2.1 Редактирование /etc(exports . . . . .	7
2.2 Настройка firewall для NFS . . . . .	7
2.3 Проверка showmount . . . . .	8
2.4 Список служб rpc . . . . .	9
2.5 Монтирование NFS . . . . .	9
2.6 Редактирование /etc/fstab . . . . .	10
2.7 remote-fs.target активен . . . . .	11
2.8 Запись в /etc(exports . . . . .	12
2.9 Запись в /etc/fstab . . . . .	12
2.10 Команды экспорта и повторная проверка . . . . .	13
2.11 Просмотр каталога на клиенте . . . . .	13
2.12 Создание каталога common и файла . . . . .	14
2.13 Добавление экспорта в /etc(exports . . . . .	14
2.14 Запись в /etc/fstab . . . . .	15
2.15 Выполнение команд экспорта . . . . .	15
2.16 Создание файлов на клиенте . . . . .	16
2.17 Отображение изменений на сервере . . . . .	16
2.18 Копирование конфигурационных файлов . . . . .	17
2.19 Содержание nfs.sh на сервере . . . . .	18
2.20 Содержание nfs.sh на клиенте . . . . .	18

# **Список таблиц**

# **1 Введение**

## **1.1 Цель работы**

Приобретение навыков настройки сервера NFS для удалённого доступа к курсам.

## **2 Процесс работы**

### **2.1 Настройка сервера NFSv4**

На сервере было установлено программное обеспечение для работы NFS. После завершения установки появились зависимости, необходимые для запуска RPC-служб.

Был создан каталог, который планировалось использовать как корневой для экспортируемого NFS-дерева.

В файл `/etc(exports` была добавлена строка, предоставляющая общий доступ ко каталогу только для чтения всем клиентам сети.

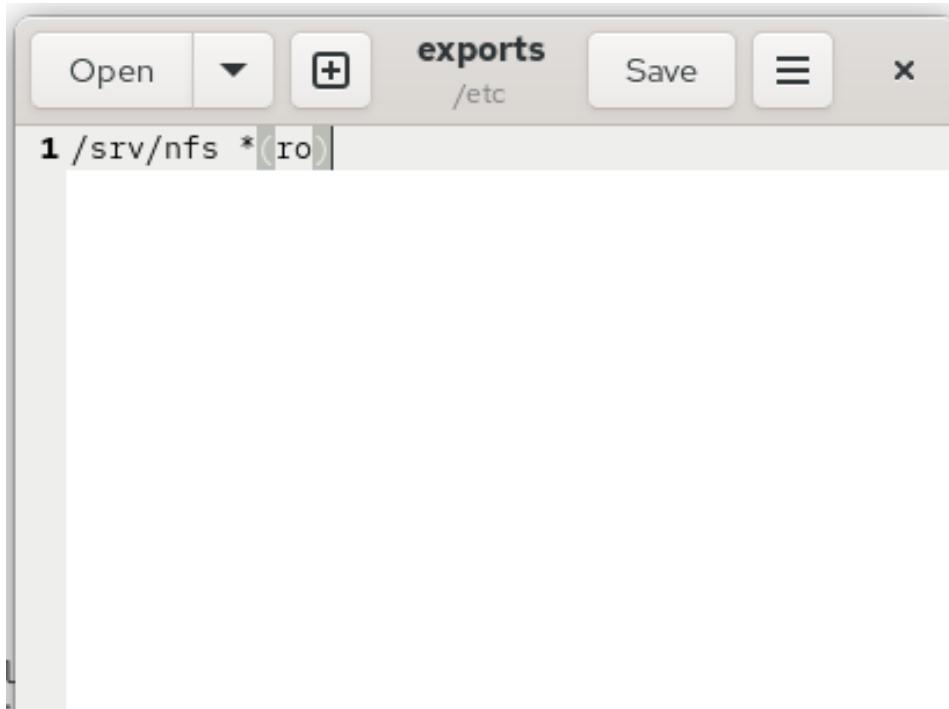


Рис. 2.1: Редактирование /etc/exports

Для каталога был назначен контекст безопасности типа `nfs_t`, после чего изменения применены ко всей структуре.

Служба NFS была запущена и добавлена в автозагрузку.

В межсетевой экран была добавлена служба NFS для разрешения входящих запросов. После изменения конфигурации firewall был перезагружен.

```
Installed:
gsproxy-0.9.2-10.el10.x86_64          libev-4.33-14.el10.x86_64          libnfsidmap-1:2.8.3-0.el10.x86_64
liberto-libev-0.3.2-10.el10.x86_64      nfs-utils-1:2.8.3-0.el10.x86_64      rpcbind-1.2.7-3.el10.x86_64
sssd-nfs-idmap-2.11.1-2.el10.1.x86_64

Complete!
[root@server.dgavdadaev.net ~]# mkdir -p /srv/nfs
[root@server.dgavdadaev.net ~]# gedit /etc/exports
[root@server.dgavdadaev.net ~]# semanage fcontext -a -t nfs_t "/srv/nfs(/.*)?"
[root@server.dgavdadaev.net ~]# restorecon -vR /srv/nfs
Relabeled /srv/nfs from unconfined_u:object_r:var_t:s0 to unconfined_u:object_r:nfs_t:s0
[root@server.dgavdadaev.net ~]# systemctl start nfs-server.service
[root@server.dgavdadaev.net ~]# systemctl enable nfs-server.service
Created symlink /etc/systemd/system/multi-user.target.wants/nfs-server.service → '/usr/lib/systemd/system/nfs-server.service'.
[root@server.dgavdadaev.net ~]# firewall-cmd --add-service=nfs
success
[root@server.dgavdadaev.net ~]# firewall-cmd --add-service=nfs --permanent
success
[root@server.dgavdadaev.net ~]# firewall-cmd --reload
success
[root@server.dgavdadaev.net ~]#
```

Рис. 2.2: Настройка firewall для NFS

После установки клиентского ПО NFS была выполнена проверка доступных

экспортов сервера.

Команда просмотра экспортов опрашивает RPC-службу mountd на сервере. При первоначальном запросе ответ не был получен из-за ограничений межсетевого экрана, но после его настройки ресурсы стали доступны.

```
sssd-proxy-2.11.1-2.el10_1.1.x86_64
Installed:
gssproxy-0.9.2-10.el10.x86_64           libev-4.33-14.el10.x86_64
libnfsidmap-1:2.8.3-0.el10.x86_64       libverto-libev-0.3.2-10.el10.x86_64
nfs-utils-1:2.8.3-0.el10.x86_64          rpcbind-1.2.7-3.el10.x86_64
sssd-nfs-idmap-2.11.1-2.el10_1.1.x86_64

Complete!
[root@client.dgavdadaev.net ~]# showmount -e server.dgavdadaev.net
clnt_create: RPC: Unable to receive
[root@client.dgavdadaev.net ~]# showmount -e server.dgavdadaev.net
Export list for server.dgavdadaev.net:
/srv/nfs *
[root@client.dgavdadaev.net ~]#
[root@client.dgavdadaev.net ~]# showmount -e server.dgavdadaev.net
Export list for server.dgavdadaev.net:
/srv/nfs *
[root@client.dgavdadaev.net ~]#
```

Рис. 2.3: Проверка showmount

После остановки службы firewall клиент смог получить список экспортов без ограничений. Это произошло потому, что все порты RPC-служб стали доступны напрямую.

Межсетевой экран был запущен вновь для продолжения настройки.

На стороне сервера была выполнена проверка активных TCP- и UDP-подключений, в которых участвовали процессы NFS, mountd и rpcbind. Эти службы обеспечивают обработку запросов на подключение и монтирование.

```

way:root@... ~]# firewall-cmd --get-services
0-AD RH-Satellite-6 capsule afp alvz amanda-client amanda-k5-client amqp amqps anno-1602 anno-1800 apcupsd aseqnet audit a
usweisapp2 bacula bacula-client bareos-director bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-testnet bitcoin-testn
et-ntp bittorrent-lsd ceph ceph-exporter ceph-mon cfengine civilization-iv civilization-v cockpit collectd condor-collector
cratedb ctdb ddd dds-multicast dds-unicast dhcp dhcpv6-client distcc dns dns-over-quic dns-over-tls docker-registry docker-swarm
dropbox-lansync elasticsearch etcd-client etcd-server factorio finger foreman foreman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-
replication freeipa-trust ftp galera ganglia-client ganglia-master git gpgsql grafana greti high-availability http http3 https ident imap ima
ps iperf2 iperf3 ipfs ipp ipp-client ipsec irc ircs icssi-target iiscs jenkins kadmin kdeconnect kerberos kibana klogind kpasswd kprop kshe
ll kube-api kube-apiserver kube-control-plane kube-control-plane-secure kube-controller-manager-secure kube-nodepe
ort-services kube-scheduler kube-scheduler-secure kube-worker kubelet kubelet-readyonly kubelet-worker ldap ldaps libvirt libvirt-tls ligh
tning-network llmnr llmnr-client llmnr-tcp llmnr-udp managesieve matrix mdns memcache minecraft mindlin mndp mongodb mosh mountd mpd mqtt
t mqtt-tls ms-wbs mssql murmur mysql nbd nebula need-for-speed-most-wanted netbios-ns netdata-dashboard nfs nfs3 nmea-0183 ntp nut o
pentellemetry openvpn ovirt-imagestore ovirt-storageconsole ovirt-vmconsole plesk pmcd pmproxy pmwebapi pop3 pop3s postgresql privox
y prometheus prometheus-node-exporter proxy-dhcp ps2link ps3netsrv ptp pulseaudio puppetmaster quassel radius radsec rdp redis redis-sent
inel rootd rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-client samba-dc sane settlers-history-collection sip sips slimevr slp
smtp smtp-submission smptps snmp snmptrap snmptrap spiderOak-lansync spotify-sync squid ssdp ssh ssh-custom statsrv steam-lan
-transfer steam-streaming stellaris stronghold-crusader stut stuns submission supertuxkart svdrp svn syncthing syncthing-gui syncthing-re
lay synergy syscomlansyslog syslog-tls telnet tentacle terraria tf2 tft tile38 tinc tor-socks turnturn turns upnp-client vd
sm vnc-server vrpp warpinator wbem-http wbem-https wireguard ws-discovery ws-discovery-client ws-discovery-host ws-discovery-tcp ws-disco
very-udp wsdd-wsdd-https wsmn wsman xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-java-gateway zabbix-server za
abbix-trapper zabbix-web-service zero-k zerotier
[root@server.dgavdadaev.net ~]# firewall-cmd --add-service=mountd --add-service=rpc-bind
success
[root@server.dgavdadaev.net ~]# firewall-cmd --add-service=mountd --add-service=rpc-bind --permanent
success
[root@server.dgavdadaev.net ~]# firewall-cmd --reload
success
[root@server.dgavdadaev.net ~]#

```

Рис. 2.4: Список служб grcs

После этого службы mountd и rpc-bind были добавлены в настройки межсетевого экрана, чтобы обеспечить корректную работу протокола NFSv4 через firewall.

## 2.2 Мониторинг NFS на клиенте

На клиентской машине был создан каталог для подключения удалённого ресурса. После этого NFS-дерево сервера было успешно подключено.

```

[root@client.dgavdadaev.net ~]#
[root@client.dgavdadaev.net ~]# mkdir -p /mnt/nfs
[root@client.dgavdadaev.net ~]# mount server.dgavdadaev.net:/srv/nfs /mnt/nfs
[root@client.dgavdadaev.net ~]# mount | grep nfs
server.dgavdadaev.net:/srv/nfs on /mnt/nfs type nfs4 (rw,relatime,vers=4.2,rsize=262144,wszie=262144,namelen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=192.168.1.30,local_lock=none,addr=192.168.1.1)
[root@client.dgavdadaev.net ~]#

```

Рис. 2.5: Мониторинг NFS

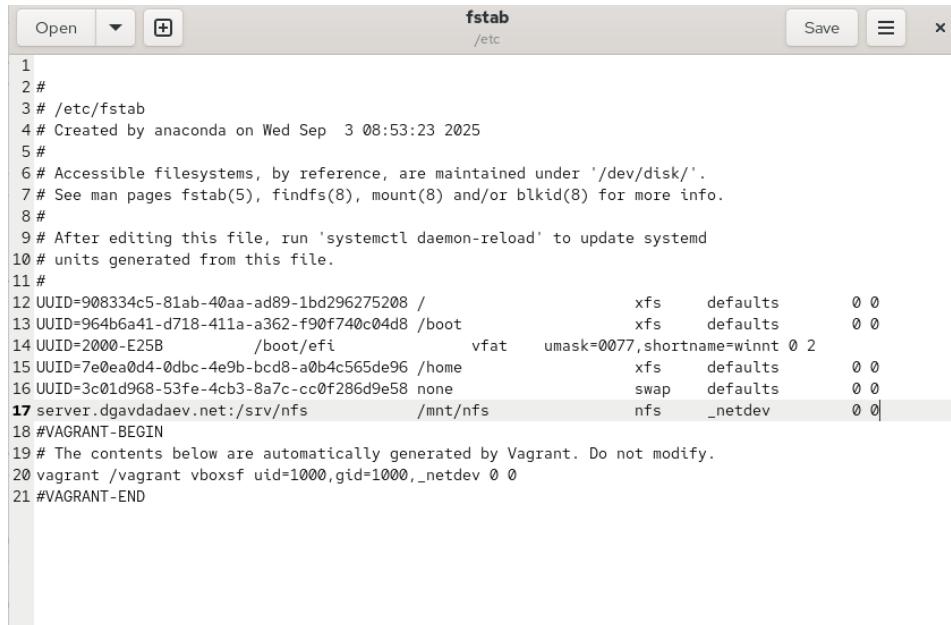
В выводе информации о мониторинге видно:

- удалённый ресурс подключён по протоколу NFSv4;
- используется транспорт TCP;

- версия NFS – 4.2;
- указаны параметры производительности и безопасности;
- отображаются адреса сервера и клиента.

Это подтверждает корректное монтирование удалённого каталога.

В конец файла `/etc/fstab` была добавлена строка, обеспечивающая автоматическое подключение ресурса при загрузке системы.



```

1
2 #
3 # /etc/fstab
4 # Created by anaconda on Wed Sep  3 08:53:23 2025
5 #
6 # Accessible filesystems, by reference, are maintained under '/dev/disk/'.
7 # See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
8 #
9 # After editing this file, run 'systemctl daemon-reload' to update systemd
10 # units generated from this file.
11 #
12 UUID=908334c5-81ab-40aa-ad89-1bd296275208 / xfs defaults 0 0
13 UUID=964b6a41-d718-411a-a362-f90f740c04d8 /boot xfs defaults 0 0
14 UUID=2000-E25B /boot/efi vfat umask=0077,shortname=winnt 0 2
15 UUID=7e0ea0d4-0dbc-4e9b-bcd8-a0b4c565de96 /home xfs defaults 0 0
16 UUID=3c01d968-53fe-4cb3-8a7c-cc0f286d9e58 none swap defaults 0 0
17 server.dgavdadaev.net:/srv/nfs /mnt/nfs nfs _netdev 0 0
18 #VAGRANT-BEGIN
19 # The contents below are automatically generated by Vagrant. Do not modify.
20 vagrant /vagrant vboxsf uid=1000,gid=1000,_netdev 0 0
21 #VAGRANT-END

```

Рис. 2.6: Редактирование `/etc/fstab`

- `server.dgavdadaev.net:/srv/nfs` – путь к удалённому NFS-каталогу;
- `/mnt/nfs` – локальная точка монтирования;
- `nfs` – тип файловой системы;
- `_netdev` – флаг, указывающий системе монтировать ресурс только после поднятия сетевых интерфейсов;
- завершающие нули отключают проверку файловой системы и резервное копирование.

Служба `remote-fs.target` была активна, что подтверждает работу механизма автоматического монтирования сетевых ресурсов.

После перезагрузки клиентской системы NFS-каталог был автоматически подключён.

```
[root@client.dgavdadaev.net ~]# systemctl status remote-fs.target
● remote-fs.target - Remote File Systems
    Loaded: loaded (/usr/lib/systemd/system/remote-fs.target; enabled; preset: enabled)
      Active: active since Sat 2025-12-06 08:13:55 UTC; 14min ago
        Invocation: c34459e4f2ef41408c8afad25c3b7648
      Docs: man:systemd.special(7)

Dec 06 08:13:55 client.dgavdadaev.net systemd[1]: Reached target remote-fs.target - Remote File System
[root@client.dgavdadaev.net ~]#
[root@client.dgavdadaev.net ~]# mount -a
[root@client.dgavdadaev.net ~]# mount | grep nfs
server.dgavdadaev.net:/srv/nfs on /mnt/nfs type nfs4 (rw,relatime,vers=4.2,rsize=262144,wsize=262144,namelen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=192.168.1.30,local_lock=none,addr=192.168.1.1)
[root@client.dgavdadaev.net ~]#
```

Рис. 2.7: remote-fs.target активен

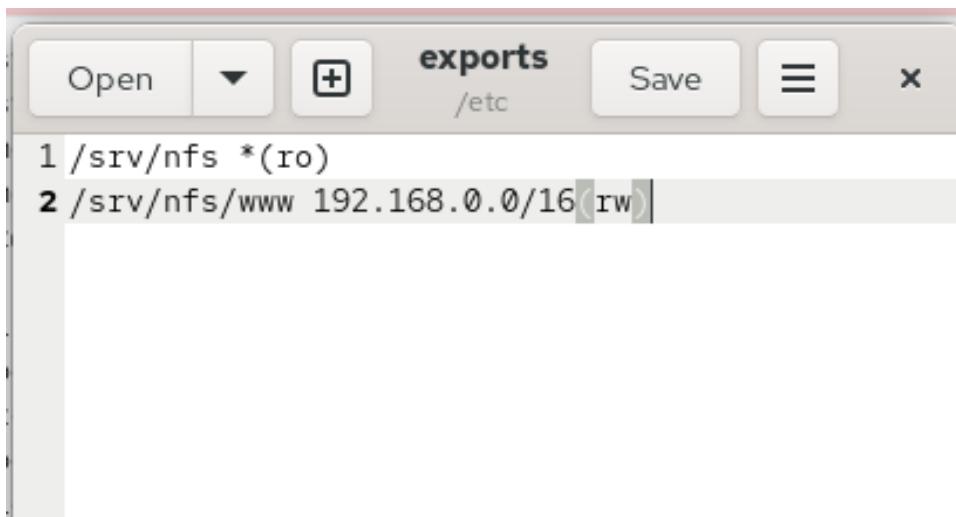
## 2.3 Подключение каталогов к дереву NFS

На сервере был создан каталог, предназначенный для публикации веб-контента через NFS. После его создания в него был подключен каталог веб-сервера с использованием механизма bind. Это позволило отобразить содержимое /var/www внутри структуры /srv/nfs.

Проверка содержимого директории /srv/nfs показала наличие каталога www, что подтверждает корректность операции.

После монтирования NFS-ресурса клиентская система корректно отображала каталог www внутри точки монтирования /mnt/nfs. Это подтверждало, что клиент получает структуру каталогов с сервера без ошибок.

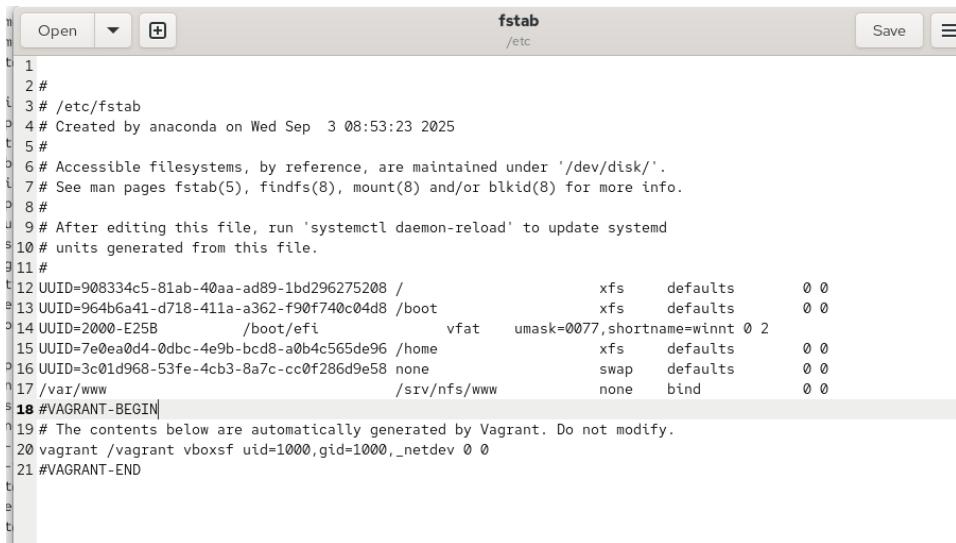
В файл /etc(exports была добавлена строка, предоставляющая доступ к каталогу веб-сервера всем машинам подсети 192.168.0.0/16 с правом записи. После обновления конфигурации экспортов сервер повторно экспортировал каталоги.



```
1 /srv/nfs *(ro)
2 /srv/nfs/www 192.168.0.0/16(rw)
```

Рис. 2.8: Запись в /etc/exports

Чтобы обеспечить автоматическое подмонтирование каталога веб-сервера в структуру дерева NFS при запуске операционной системы, в конец файла /etc/fstab была добавлена строка с типом bind.



```
1
2 #
3 # /etc/fstab
4 # Created by anaconda on Wed Sep  3 08:53:23 2025
5 #
6 # Accessible filesystems, by reference, are maintained under '/dev/disk/'.
7 # See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
8 #
9 # After editing this file, run 'systemctl daemon-reload' to update systemd
10 # units generated from this file.
11 #
12 UUID=908334c5-81ab-40aa-ad89-1bd296275208 / xfs defaults 0 0
13 UUID=964b6a41-d718-411a-a362-f90f740c04d8 /boot xfs defaults 0 0
14 UUID=2000-E25B /boot/efi vfat umask=0077,shortname=winnt 0 2
15 UUID=7@ea0d4-0dbc-4e9b-bcd8-a0b4c565de96 /home xfs defaults 0 0
16 UUID=3c01d968-53fe-4cb3-8a7c-cc0f286d9e58 none swap defaults 0 0
17 /var/www /srv/nfs/www none bind 0 0
18 #VAGRANT-BEGIN
19 # The contents below are automatically generated by Vagrant. Do not modify.
20 vagrant /vagrant vboxsf uid=1000,gid=1000,_netdev 0 0
21 #VAGRANT-END
```

Рис. 2.9: Запись в /etc/fstab

После обновления конфигурации каталоги были повторно экспортированы. Клиент после этого продолжал корректно видеть каталог www внутри своей точки монтирования /mnt/nfs.

```
[root@server.dgavdadaev.net ~]#  
[root@server.dgavdadaev.net ~]# mkdir -p /srv/nfs/www  
[root@server.dgavdadaev.net ~]# mount -o bind /var/www/ /srv/nfs/www/  
[root@server.dgavdadaev.net ~]# ls /srv/nfs/  
www  
[root@server.dgavdadaev.net ~]# gedit /etc/exports  
[root@server.dgavdadaev.net ~]# exportfs -r  
[root@server.dgavdadaev.net ~]# gedit /etc/fstab  
[root@server.dgavdadaev.net ~]# exportfs -r  
[root@server.dgavdadaev.net ~]# mount -a  
[root@server.dgavdadaev.net ~]# █
```

Рис. 2.10: Команды экспорта и повторная проверка

Финальная проверка на клиентской машине показала, что каталог `WWW` доступен и корректно отображен в дереве NFS, что подтверждает успешную настройку как bind-монтирования, так и экспорта каталога по сети.

```
[root@client.dgavdadaev.net ~]#  
[root@client.dgavdadaev.net ~]# ls /mnt/nfs/  
www  
[root@client.dgavdadaev.net ~]# ls /mnt/nfs/  
www  
[root@client.dgavdadaev.net ~]#  
[root@client.dgavdadaev.net ~]# ls /mnt/nfs/  
www  
[root@client.dgavdadaev.net ~]# █
```

Рис. 2.11: Просмотр каталога на клиенте

## 2.4 Подключение каталогов для работы пользователей

Под пользователем `dgavdadaev` в его домашнем каталоге был создан каталог `common` с правами доступа 700. Внутри каталога создан файл `dgavdadaev@server.txt`.

```
[dgavdadaev@server.dgavdadaev.net ~]$ mkdir -p -m 700 ~/common
[dgavdadaev@server.dgavdadaev.net ~]$ cd common/
[dgavdadaev@server.dgavdadaev.net common]$ touch dgavdadaev@server.txt
[dgavdadaev@server.dgavdadaev.net common]$ ls
dgavdadaev@server.txt
[dgavdadaev@server.dgavdadaev.net common]$
```

Рис. 2.12: Создание каталога common и файла

Права доступа каталога и файлов ограничивают доступ только пользователю. На сервере был подготовлен каталог для экспорта домашнего каталога пользователя:



Рис. 2.13: Добавление экспорта в /etc/exports

В файл /etc/fstab была добавлена строка для автоматического bind-монтирования каталога пользователя в дерево NFS:

```

1
2 #
3 # /etc/fstab
4 # Created by anaconda on Wed Sep  3 08:53:23 2025
5 #
6 # Accessible filesystems, by reference, are maintained under '/dev/disk/'.
7 # See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
8 #
9 # After editing this file, run 'systemctl daemon-reload' to update systemd
10 # units generated from this file.
11 #
12 UUID=908334c5-81ab-40aa-ad89-1bd296275208 /          xfs    defaults      0 0
13 UUID=964b6a41-d718-411a-a362-f90f740c04d8 /boot      xfs    defaults      0 0
14 UUID=2000-E25B          /boot/efi        vfat    umask=0077,shortname=wintnt 0 2
15 UUID=7e0ea0d4-0dbc-4e9b-bcd8-a0b4c565de96 /home      xfs    defaults      0 0
16 UUID=3c01d968-53fe-4cb3-8a7c-cc0f286d9e58 none      swap    defaults      0 0
17 /var/www              /srv/nfs/www     none    bind         0 0
18 /home/dgavdadaev/common /srv/nfs/home/dgavdadaev none    bind         0 0
19 #VAGRANT-BEGIN
20 # The contents below are automatically generated by Vagrant. Do not modify.
21 vagrant /vagrant vboxsf uid=1000,gid=1000,_netdev 0 0
22 #VAGRANT-END

```

Рис. 2.14: Запись в /etc/fstab

На сервере были созданы каталоги, выполнено bind-монтирование и обновлены экспорты:

```

[root@server dgavdadaev.net ~]# mount -a
[root@server dgavdadaev.net ~]# mkdir -p /srv/nfs/home/dgavdadaev
[root@server dgavdadaev.net ~]# mount -o bind /home/dgavdadaev/common /srv/nfs/home/dgavdadaev/
mount: (hint) your fstab has been modified, but systemd still uses
      the old version; use 'systemctl daemon-reload' to reload.
[root@server dgavdadaev.net ~]# gedit /etc/exports
[root@server dgavdadaev.net ~]# gedit /etc/fstab
[root@server dgavdadaev.net ~]# exportfs -r
[root@server dgavdadaev.net ~]#

```

Рис. 2.15: Выполнение команд экспорта

На клиенте каталог пользователя корректно отображается внутри /mnt/nfs/home/dgavdadaev/

В каталоге созданы файлы dgavdadaev@client.txt и dgavdadaev@client.txt от разных пользователей.

```
[dgavdadaev@client.dgavdadaev.net ~]$  

[dgavdadaev@client.dgavdadaev.net ~]$ cd /mnt/nfs/home/dgavdadaev/  

[dgavdadaev@client.dgavdadaev.net dgavdadaev]$ ls  

dgavdadaev@server.txt  

[dgavdadaev@client.dgavdadaev.net dgavdadaev]$ touch dgavdadaev@client.txt  

[dgavdadaev@client.dgavdadaev.net dgavdadaev]$ ls  

dgavdadaev@client.txt dgavdadaev@server.txt  

[dgavdadaev@client.dgavdadaev.net dgavdadaev]$ ls -l  

total 0  

-rw-r--r-- 1 dgavdadaev dgavdadaev 0 Dec 6 08:38 dgavdadaev@client.txt  

-rw-r--r-- 1 dgavdadaev dgavdadaev 0 Dec 6 08:34 dgavdadaev@server.txt  

[dgavdadaev@client.dgavdadaev.net dgavdadaev]$ sudo -i  

[sudo] password for dgavdadaev:  

[root@client.dgavdadaev.net ~]# cd /mnt/nfs/home/dgavdadaev/  

-bash: cd: /mnt/nfs/home/dgavdadaev/: Permission denied  

[root@client.dgavdadaev.net ~]# █
```

Рис. 2.16: Создание файлов на клиенте

Права доступа позволяют пользователю dgavdadaev создавать и изменять файлы.

При попытке перейти в каталог под root возникло ограничение доступа, что связано с правами 700:

После создания файлов на клиенте соответствующие изменения появились на сервере в каталоге пользователя:

```
[dgavdadaev@server.dgavdadaev.net ~]$  

[dgavdadaev@server.dgavdadaev.net ~]$ mkdir -p -m 700 ~/common  

[dgavdadaev@server.dgavdadaev.net ~]$ cd  

[dgavdadaev@server.dgavdadaev.net ~]$ cd common/  

[dgavdadaev@server.dgavdadaev.net common]$ touch dgavdadaev@server.txt  

[dgavdadaev@server.dgavdadaev.net common]$ ls  

dgavdadaev@server.txt  

[dgavdadaev@server.dgavdadaev.net common]$  

[dgavdadaev@server.dgavdadaev.net common]$ ls -l  

total 0  

-rw-r--r-- 1 dgavdadaev dgavdadaev 0 Dec 6 08:38 dgavdadaev@client.txt  

-rw-r--r-- 1 dgavdadaev dgavdadaev 0 Dec 6 08:34 dgavdadaev@server.txt  

[dgavdadaev@server.dgavdadaev.net common]$ █
```

Рис. 2.17: Отображение изменений на сервере

## 2.5 Внесение изменений в настройки внутреннего окружения виртуальных машин

В каталоге `/vagrant/provision/server/` был создан подкаталог `nfs/etc` и в него скопирован файл `/etc/exports`.

```
[root@server.dgavdadaev.net ~]#  
[root@server.dgavdadaev.net ~]# cd /vagrant/provision/server/  
[root@server.dgavdadaev.net server]# mkdir -p /vagrant/provision/server/nfs/etc  
[root@server.dgavdadaev.net server]# cp -R /etc/exports /vagrant/provision/server/nfs/etc/  
[root@server.dgavdadaev.net server]# touch nfs.sh  
[root@server.dgavdadaev.net server]#
```

Рис. 2.18: Копирование конфигурационных файлов

В каталоге `/vagrant/provision/server/` создан файл `nfs.sh`.

Внутри файла прописан скрипт автоматической настройки NFS, включая:

- установку пакетов,
- копирование конфигурации,
- настройку SELinux,
- настройку firewall,
- подготовку каталогов `/srv/nfs/www`, `/srv/nfs/home/...`,
- bind-монтирование каталогов,
- добавление записей в `/etc/fstab`,
- запуск службы NFS.

```

1  #!/bin/bash
2  echo "Provisioning script $0"
3  echo "Install needed packages"
4  dnf -y install nfs-utils
5  echo "Copy configuration files"
6  cp -R /vagrant/provision/server/nfs/etc/* /etc
7  restorecon -vR /etc
8  echo "Configure firewall"
9  firewall-cmd --add-service nfs --permanent
10 firewall-cmd --add-service mountd --add-service rpc-bind --permanent
11 firewall-cmd --reload
12 echo "Tuning SELinux"
13 mkdir -p /srv/nfs
14 semanage fcontext -a -t nfs_t "/srv/nfs(/.*)?"
15 restorecon -vR /srv/nfs
16 echo "Mounting dirs"
17 mkdir -p /srv/nfs/www
18 mount -o bind /var/www /srv/nfs/www
19 echo "/var/www /srv/nfs/www none bind 0 0" >> /etc/fstab
20 mkdir -p /home/dgavdadaev
21 mkdir -p -m 700 /home/dgavdadaev/common
22 chown dgavdadaev:dgavdadaev /home/dgavdadaev/common
23 mount -o bind /home/dgavdadaev/common /srv/nfs/home/dgavdadaev
24 echo "/home/dgavdadaev/common /srv/nfs/home/dgavdadaev none bind 0 0" >> /etc/fstab
25 echo "Start nfs service"
26 systemctl enable nfs-server
27 systemctl start nfs-server
28 systemctl restart firewalld
29

```

Рис. 2.19: Содержание nfs.sh на сервере

В каталоге `/vagrant/provision/client/` был создан исполняемый файл `nfs.sh`, предназначенный для автоматической настройки NFS-клиента, включая:

- установку пакетов NFS,
- создание точки монтирования,
- монтирование каталога,
- добавление записи в `/etc/fstab`,
- восстановление SELinux-контекстов.



```

Vagrantfile nfs.sh nfs.sh ✘
1  #!/bin/bash
2  echo "Provisioning script $0"
3  echo "Install needed packages"
4  dnf -y install nfs-utils
5  echo "Mounting dirs"
6  mkdir -p /mnt/nfs
7  mount server.dgavdadaev.net:/srv/nfs /mnt/nfs
8  echo "server.dgavdadaev.net:/srv/nfs /mnt/nfs nfs _netdev 0 0" >> /etc/fstab
9  restorecon -vR /etc

```

Рис. 2.20: Содержание nfs.sh на клиенте

# **3 ИТОГИ**

## **3.1 Вывод**

В ходе работы были настроены экспортируемые каталоги NFS, включая системные и пользовательские ресурсы. Реализовано bind-монтирование каталогов веб-сервера и домашнего каталога пользователя. На клиенте выполнена проверка доступа к удалённым ресурсам, обеспечена возможность создания файлов, а также изучено поведение при различных правах доступа. Кроме того, подготовлены provisioning-скрипты, автоматизирующие настройку NFS-сервера и клиента. Лабораторная работа успешно завершена, все пункты задания выполнены.

## **3.2 Контрольные вопросы**

### **1. Как называется файл конфигурации, содержащий общие ресурсы NFS?**

Файл `/etc(exports`, в котором перечислены каталоги, доступные клиентам NFS, а также параметры их экспорта.

### **2. Какие порты должны быть открыты в брандмауэре, чтобы обеспечить полный доступ к серверу NFS?**

Необходимо открыть службы NFS, RPC-bind и mountd.

Они используют следующие порты:

- NFSv4 – TCP 2049
- rpcbind – TCP/UDP 111

- mountd – динамический порт (обычно задаётся в конфигурации или открывается как сервис в firewall)

**3. Какую опцию следует использовать в /etc/fstab, чтобы убедиться, что общие ресурсы NFS могут быть установлены автоматически при перезагрузке?**

Опция \_netdev, указывающая системе монтировать ресурс только после запуска сетевых интерфейсов, что необходимо для корректного автоматического подключения NFS.